

# Contenido

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Objetivo . . . . .	3
1.1.1	Objetivos específicos . . . . .	4
1.2	Metodología . . . . .	4
1.2.1	pyFEM . . . . .	4
	<b>Bibliografía</b>	<b>9</b>



# 1 Introducción

Los programas de computador comerciales para el análisis y diseño de estructuras que se encuentran vigentes a la fecha cuentan, en general, con un entorno gráfico que le permite al usuario describir el modelo de forma interactiva, procesarlo y visualizar los resultados de manera conveniente.

En Escamilla, 1995 se presenta una lista de algunos de estos programas de uso común en América Latina, entre los cuales se encuentra *ETABS* (Three Dimensional Analysis of Building Systems - Extended Version).

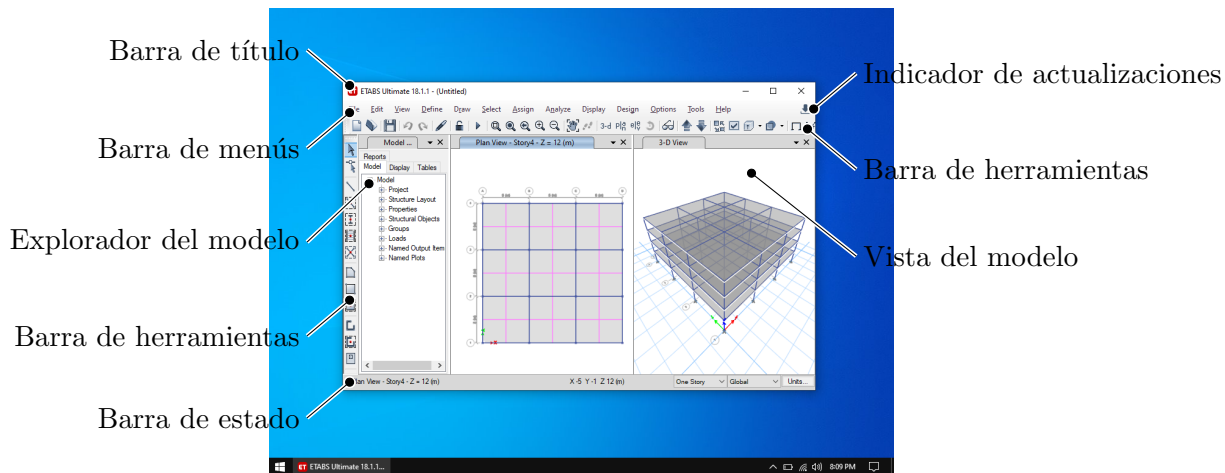
ETABS es un programa de computador creado por Edward Wilson, Jeffery Hollings y Henry Dovey en 1975. Según Wilson et al, 1975, este programa de computador fue desarrollado para el análisis estructural lineal de edificios de pórticos y muros a cortante sujetos tanto a cargas estáticas como sísmicas. El edificio es idealizado como un sistema de elementos tipo pórticos y muros a cortante independientes interconectado por losas de entrepiso las cuales son rígidas en su propio plano.

Este programa es una extensión de *TABS* (Three Dimensional Analysis of Building Systems) para poder analizar pórticos en tres dimensiones. Según Wilson y Dovey, 1972, una de las razones para desarrollar TABS fue darle una retroalimentación a los usuarios de los programas *FRMSTC* (Static Load Analysis of High-Rise Buildings), *FRMDYN* (Dynamic Analysis of Multistory Buildings), *LATERAL* y *SOLID SAP* (Static Analysis Program for Three-Dimensional Solid Structures).

FRMSTC permitía analizar edificios simétricos con pórticos y muros a cortante paralelos sujetos a cargas estáticas y evaluar los modos y las frecuencias. FRMDYN era similar a FRMSTC con la excepción que la carga era la aceleración del terreno debido a un desplazamiento dependiente del tiempo. LATERAL fue una extensión de FRMSTC que permitía analizar linealmente pórticos y muros a cortante que no eran necesariamente paralelos con tres grados de libertad en cada piso. SOLID SAP era un programa general de elementos finitos y tenía una opción que permitía introducir la aproximación de piso rígido. Este programa también tenía la opción de realizar análisis dinámico.

En la actualidad, ETABS se encuentra en la versión 18.1.1 y según Computers y Structures,

2020, puede ser ejecutado en computadores con sistema operativo Windows 7, Windows 8 o Windows 10 con arquitectura de 64 bits que cuenten como mínimo con un procesador Intel Pentium 4 o AMD Athlon 64, una resolución de 1024x768 pixeles con 16 bits por canal, 8 GB de RAM y 6 GB de espacio en el disco duro. En la figura 1-1 se presenta la ventana del programa ETABS ejecutandose en un computador con Windows 10.



**Figura 1-1:** Ventana del programa ETABS ejecutandose en Windows 10.

A través de múltiples cuadros de diálogo, los cuales son accesibles ya sea a través de la barra de menús, las barras de herramientas, el explorador del modelo, las vistas del modelo o con atajos de teclado, el usuario es capaz de modelar la estructura que desea analizar al describir los materiales, las secciones transversales, los elementos estructurales, las condiciones de apoyo, los diafragmas y las cargas.

Según Computers y Structures, 2017, ETABS analiza el modelo usando el motor de análisis *SAPFire*, el cual es común a otros programas de la misma compañía (*SAP2000*, *SAFE* y *CSiBridge*). *SAPFire* es la última versión de la serie de programas *SAP* y ofrece las siguientes herramientas:

- Análisis estático y dinámico,
- Análisis lineal y no lineal,
- Análisis sísmico y análisis incremental no lineal (*pushover*),
- Análisis de cargas móviles,
- No linealidad geométrica, incluyendo efectos P-delta y grandes desplazamientos,
- Etapas constructivas,

- Fluencia lenta (*creep*), retracción (*shrinkage*) y envejecimiento,
- Análisis de pandeo,
- Análisis de densidad espectral de potencia y estado estacionario,
- Elementos tipo pórtico y laminares, incluyendo el comportamiento de vigas, columnas, cerchas, membranas y placas,
- Elementos tipo cable y tendón,
- Elementos bidimensionales planos y elementos sólidos asimétricos,
- Elementos sólidos tridimensionales,
- Resortes no lineales y apoyos,
- Propiedades de los resortes y apoyos dependientes de la frecuencia,

Con los resultados del análisis del modelo, el posprocesador de ETABS puede *diseñar* los elementos estructurales de acuerdo a uno de varios códigos de diseño de diferentes países. ETABS es capaz de diseñar pórticos en acero, pórticos en concreto, vigas compuestas, columnas compuestas, vigas en acero de alma abierta (*steel joist*), muros a cortante y losas de concreto.

Adicionalmente, según Computers y Structures, 2019, ETABS cuenta con la posibilidad de generar dibujos estructurales esquemáticos de las plantas estructurales, de los despieces de vigas, columnas y muros a cortante, y de los detalles de las conexiones de acero.

En términos generales, estos programas de computador comerciales cuenta con características similares a las de ETABS. Actualmente, dichos programas están innovando para permitirle al usuario trabajar con modelos *BIM* (Building Information Modeling) con la intención de centralizar toda la información del proyecto y así obtener los beneficios que trae este esquema de trabajo.

## 1.1. Objetivo

Desarrollar un programa de computador a código abierto para el análisis de estructuras tridimensionales tipo pórtico sometidas a cargas estáticas.

Con este trabajo se pretende contribuir al ejercicio libre de la profesión del ingeniero estructural y a la enseñanza del análisis de las estructuras.

### 1.1.1. Objetivos específicos

- Desarrollar el ambiente gráfico y la interfaz gráfica de usuario del programa de computador para permitirle al usuario ingresar los datos que describen la estructura, las acciones a las cuales se encuentra sometida y visualizar los resultados del análisis estructural.
- Desarrollar el módulo de análisis estructural para calcular el desplazamiento de los nudos, el valor de las reacciones y de las fuerzas internas de los elementos de una estructura sometida a cargas estáticas.

## 1.2. Metodología

Se desarrollaron los programas de computador *pyFEM* y *FEM.js*. El primero para analizar estructuras tridimensionales tipo pórtico sometidas a cargas estáticas y el segundo para modelarlas. Esto con el fin que FEM.js pueda ser usado junto con otro programa de computador diferente a pyFEM.

Durante el desarrollo de estos programas se utilizó *git* como sistema de control de versiones. Según Chacon, 2014, git, creado en 2005 por Linus Torvalds, es un sistema distribuido de control de versiones que registra los cambios realizados a un conjunto de archivos para coordinar el trabajo entre programadores.

Una copia de los repositorios de pyFEM y FEM.js se encuentran en la página de internet *GitHub*, la cual permite alojar proyectos utilizando git. pyFEM está alojado en <https://github.com/rvcristiand/pyFEM> mientras que FEM.js está alojado en <https://github.com/rvcristiand/FEM.js>.

### 1.2.1. pyFEM

pyFEM fue desarrollado en *Python*. Según Lutz, 2013, Python es un lenguaje de programación interpretado orientado a objetos cuya filosofía hace énfasis en la legibilidad de su código. Los archivos revelantes que componen el repositorio de pyFEM son:

```
pyFEM/  
├── LICENSE  
├── README.md  
├── example_1.json  
├── example_2.json  
├── example_3.json  
├── pyFEM/  
│   ├── classtools.py  
│   ├── core.py  
│   └── primitives.py  
└── test/  
    ├── space_frame.py  
    └── trusses.py
```

El archivo `LICENCE` contiene la licencia de pyFEM, la cual transcribo a continuación.

MIT License

Copyright (c) 2019 pyFEM

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

El archivo `README.md` contiene todas las instrucciones necesarias para ejecutar y usar pyFEM.

Los archivos `example_1.json`, `example_2.json` y `example_3.json` almacenan los modelos de tres de los ejemplos presentados en Escamilla, 1995 que han sido creados con pyFEM con el proposito de validar los resultados del programa.

La extensión `json` se usa para indicar que los archivos tienen formato *JSON* (de sus siglas en inglés JavaScript Object Notation) el cual es un formato sencillo para el intercambio de datos. A continuación se presenta la estructura general que tienen estos archivos.

```
{
    "materials": {
        "1": {
            "E": 20400000.0,
            "G": 0
        },
        ...
    },
    "sections": {
        "1": {
            "area": 0.003,
            "Ix": 0,
            "Iy": 0,
            "Iz": 0,
            "type": "Section"
        },
        ...
    },
    "joints": {
        "1": {
            "x": 0,
            "y": 0,
            "z": 0
        },
        ...
    },
    "frames": {
        "1-3": {
            "j": 1,
            "k": 3,
            "material": "1",
            "section": "3"
        },
        ...
    },
    "supports": {
        "1": {
            "ux": true,
            "uy": true,
            "uz": false,
            "rx": false,
            "ry": false,
            "rz": false
        },
        ...
    },
    "load_patterns": {
        "point loads": {
            "joints": {
                "3": [
                    {
                        "fx": 4.0,
                        "fy": 3.0,
                        "fz": 3.0,
                        "mx": 0,
                        "my": 0,
                        "mz": 0
                    }
                ]
            },
            ...
        },
        ...
    }
}
```



```
    },  
    ...  
  }  
}
```

Los archivos son almacenados en formato *JSON* para ser leídos por FEM.js.

utilizando el paradigma de programación orientada a objetos. .

empleando técnicas y herramientas . pyFEM es un programa de computador desarrollado en *python* programas de computador

- Identificar las bibliotecas necesarias para el desarrollo del programa de computador: se investigarán diferentes posibilidades para seleccionar el lenguaje de programación, las bibliotecas más apropiadas para el desarrollo del ambiente gráfico, la interfaz gráfica de usuario, el módulo de análisis estructural y la compatibilidad entre ellas.

# Bibliografía

- Chacon, S. (2014). *Pro Git*. Berkeley, CA New York, NY, Apress, Distributed to the Book trade worldwide by Spring Science+Business Media.
- Computers & Structures. (2017). *CSi Anlysis Reference Manual*.
- Computers & Structures. (2019). *Welcome to ETABS*.
- Computers & Structures. (2020). ETABS System Requirements [Accedido: 2020-09-29].
- Escamilla, J. (1995). *Microcomputadores en ingeniería estructural*. Santafé de Bogotá, ECOE Universidad Nacional de Colombia. Facultad de Ingeniera.
- Lutz, M. (2013). *Learning Python*. Sebastopol, CA, O'Reilly.
- Wilson, E. L. & Dovey, H. H. (1972). Three dimensional analysis of building systems - TABS. *Earthwuake engineering research center*.
- Wilson, E. L., Hollings, J. P. & Dovey, H. (1975). Three dimensional analysis of building systems (extended version). *Earthwuake engineering research center*.